

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appl. No.: 10/820,442 Confirmation No.: 7157  
Appellant(s): Mikko Makela  
Filed: April 7, 2004  
Art Unit: 2175  
Examiner: Andrew L. Tank  
Title: PRESENTATION OF LARGE PAGES ON SMALL DISPLAYS

Docket No.: 042933/378783  
Customer No.: 00826

Mail Stop Appeal Brief-Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**APPEAL BRIEF UNDER 37 CFR § 41.37**

This Appeal Brief is filed pursuant to the "Notice of Appeal to the Board of Patent Appeals and Interferences" filed April 7, 2010, and is responsive to the Final Office Action dated January 7, 2010 ("the Final Office Action"), and the Pre-Brief Appeal Conference decision dated August 20, 2010.

**Table of Contents:**

1. Real Party in Interest, begins on Page 2;
2. Related Appeals and Interferences, begins on Page 3;
3. Status of Claims, begins on page 4;
4. Status of Amendments, begins on page 5;
5. Summary of Claimed Subject Matter, begins on page 6;
6. Grounds of Rejection to be reviewed on Appeal, begins on page 9;
7. Argument, begins on page 10;
8. Claims Appendix, begins on page 15;
9. Evidence Appendix, begins on page 24; and
10. Related Proceedings Appendix, begins on page 25.

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

1.     ***Real Party in Interest.***

The real party in interest in this appeal is Nokia Corporation, the assignee of the above-referenced patent application.

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

2. ***Related Appeals and Interferences.***

There are no related appeals and/or interferences involving this application or its subject matter.

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

3. ***Status of Claims.***

The present application includes pending Claims 1, 14, 15, and 25-41, all of which stand rejected. Claims 2-13 and 16-24 were previously canceled. All of pending Claims 1, 14, 15, and 25-41 are hereby being appealed.

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

4. *Status of Amendments.*

There are no un-entered amendments in this application.

5. ***Summary of Claimed Subject Matter.***

The claimed invention will now be summarized with reference to passages of the specification. It should be understood, however, that the references are provided solely for explanatory purposes, and should not otherwise in and of themselves be taken to limit the scope of the claimed invention.

**Independent Claim 1**

Independent Claim 1 is directed to a method including partially dividing at least one page into a plurality of areas. FIGS. 5a-5c depict an HTML page that has been divided into a plurality of areas, where the areas may be scaled from a larger HTML page, for example, into a smaller page that is viewable on a smaller screen. The areas in small representation may, for example, be obtained from the original size HTML page via a process of dividing the HTML page and scaling each of the divided parts. See Paragraphs [0068] and [0069] of the published application. Further support may be found in paragraphs [0090-0095], with reference to FIG. 7, which describe an “exemplary flowchart for an algorithm for dividing one or several HTML pages into areas[.]”

Claim 1 further includes presenting said plurality of areas in a first representation. FIG. 6 depicts a flowchart of an example embodiment of the present invention in which an HTML page may be presented on the display in a first representation in operation 603, which may, for instance, be a small representation. Operation 603 may also include the transformation of the areas from the original size format to a first representation format via scaling, cropping, or using an icon. See, for example, paragraph [0063].

Independent Claim 1 further defines that at least partially dividing at least one page into a plurality of areas includes element-wise rendering elements contained in the at least one page to obtain a rendered object with a maximum height and a maximum width, checking if a size of said rendered object exceeds a threshold, and forming an area from said rendered object if the threshold is exceeded. In operation 701 of the flowchart of FIG 7, HTML elements of one or several HTML pages may be rendered and investigated in the order they appear in the HTML source code of the page or pages. A calculation of pixel values corresponding to the HTML objects may be, for instance, performed as if an HTML page was shown in its original layout

with 100% zoom. As a result, a maximum height and a maximum width in pixels of a number of rendered HTML objects may be obtained. See paragraphs [0073] and [0074]. If the product of the maximum height and width is larger than a pre-defined threshold, a rectangular area containing the HTML objects rendered in operation 701 is formed in operation 703 of the flowchart of FIG. 7. See paragraphs [0075] – [0077].

Independent Claim 1 further includes checking if at least one edge of the formed area is not straight, and forming a smaller area from the rendered object if at least one edge is not straight. In operation 705 of FIG. 7, the method checks whether the formed area would or would not have a straight top edge. If the formed area would not have a top edge, the algorithm may return to operation 703 and attempt to form a new area with a straight top edge. See at least paragraph [0081].

#### **Independent Claim 15**

Independent Claim 15 is directed to an apparatus including a processor configured to at least partially divide at least one page into a plurality of areas. FIGS. 5a-5c depict an HTML page that has been divided into a plurality of areas, where the areas may be scaled from a larger HTML page, for example, into a smaller page that is viewable on a smaller screen. The areas in small representation may, for example, be obtained from the original size HTML page via a process of dividing the HTML page and scaling each of the divided parts. See Paragraphs [0068] and [0069] of the published application. Further support may be found in paragraphs [0090-0095], with reference to FIG. 7, which describe an “exemplary flowchart for an algorithm for dividing one or several HTML pages into areas[.]”

Claim 15 further includes a display configured to present the plurality of areas in a first representation. FIG. 6 depicts a flowchart of an example embodiment of the present invention in which an HTML page may be presented on the display in a first representation in operation 603, which may, for instance, be a small representation. Operation 603 may also include the transformation of the areas from the original size format to a first representation format via scaling, cropping, or using an icon. See, for example, paragraph [0063].

Independent Claim 15 further requires the processor to be configured to make at least one of the plurality of areas an active area and the display being further configured to present at least

one of the at least one active areas in a second representation in response to a user operation on the at least one active area. Paragraph [0089] discloses that as a result of the algorithm of FIG. 7, a plurality of areas is output and these areas can be made active for display on the display in a small representation. Further, upon selection, the areas can be displayed in a large representation.

Independent Claim 15 further defines that the processor is configured to at least partially divide at least one page into a plurality of areas including element-wise rendering elements contained in the at least one page to obtain a rendered object with a maximum height and a maximum width, checking if a size of said rendered object exceeds a threshold, and forming an area from said rendered object if the threshold is exceeded. In operation 701 of the flowchart of FIG. 7, HTML elements of one or several HTML pages may be rendered and investigated in the order they appear in the HTML source code of the page or pages. A calculation of pixel values corresponding to the HTML objects may be, for instance, performed as if an HTML page was shown in its original layout with 100% zoom. As a result, a maximum height and a maximum width in pixels of a number of rendered HTML objects may be obtained. See paragraphs [0073] and [0074]. If the product of the maximum height and width is larger than a pre-defined threshold, a rectangular area containing the HTML objects rendered in operation 701 is formed in operation 703 of the flowchart of FIG. 7. See paragraphs [0075] – [0077].

Independent Claim 15 further requires the processor to be configured to check if at least one edge of the formed area is not straight, and forming a smaller area from the rendered object if at least one edge is not straight. In operation 705 of FIG. 7, the method checks whether the formed area would or would not have a straight top edge. If the formed area would not have a top edge, the algorithm may return to operation 703 and attempt to form a new area with a straight top edge. See at least paragraph [0081].

### **Independent Claim 32**

Independent Claim 32 recites a method including at least partially dividing at least one page into a plurality of areas. FIGS. 5a-5c depict an HTML page that has been divided into a plurality of areas, where the areas may be scaled from a larger HTML page, for example, into a smaller page that is viewable on a smaller screen. The areas in small representation may, for example, be obtained from the original size HTML page via a process of dividing the HTML



page and scaling each of the divided parts. See Paragraphs [0068] and [0069] of the published application. Further support may be found in paragraphs [0090-0095], with reference to FIG. 7, which describe an “exemplary flowchart for an algorithm for dividing one or several HTML pages into areas[.]”

Independent Claim 32 further includes outputting the plurality of areas for presenting the plurality of areas in a first representation, making at least one area of the plurality of areas an active area, and for presenting at least one of the active areas in a second representation in response to a user operation on the at least one active area. Paragraph [0089] discloses that as a result of the algorithm of FIG. 7, a plurality of areas is output and these areas can be made active for display on the display in a small representation. Further, upon selection, the areas can be displayed in a large representation.

Independent Claim 32 further defines the at least partially dividing at least one page into a plurality of areas to include element-wise rendering elements contained in the at least one page to obtain a rendered object with a maximum height and a maximum width, checking if a size of the rendered object exceeds a threshold, and forming an area from the rendered object if the threshold is exceeded. In operation 701 of the flowchart of FIG 7, HTML elements of one or several HTML pages may be rendered and investigated in the order they appear in the HTML source code of the page or pages. A calculation of pixel values corresponding to the HTML objects may be, for instance, performed as if an HTML page was shown in its original layout with 100% zoom. As a result, a maximum height and a maximum width in pixels of a number of rendered HTML objects may be obtained. See paragraphs [0073] and [0074]. If the product of the maximum height and width is larger than a pre-defined threshold, a rectangular area containing the HTML objects rendered in operation 701 is formed in operation 703 of the flowchart of FIG. 7. See paragraphs [0075] – [0077].

Independent Claim 32 also recites checking if at least one edge of the formed area is not straight and forming a smaller area from the rendered object if at least one edge is not straight. In operation 705 of FIG. 7, the method checks whether the formed area would or would not have a straight top edge. If the formed area would not have a top edge, the algorithm may return to

operation 703 and attempt to form a new area with a straight top edge. See at least paragraph [0081].

### **Independent Claim 37**

Independent Claim 37 recites an apparatus that includes a processor configured to at least partially divide at least one page into a plurality of areas. FIGS. 5a-5c depict an HTML page that has been divided into a plurality of areas, where the areas may be scaled from a larger HTML page, for example, into a smaller page that is viewable on a smaller screen. The areas in small representation may, for example, be obtained from the original size HTML page via a process of dividing the HTML page and scaling each of the divided parts. See Paragraphs [0068] and [0069] of the published application. Further support may be found in paragraphs [0090-0095], with reference to FIG. 7, which describe an “exemplary flowchart for an algorithm for dividing one or several HTML pages into areas[.]”

The apparatus of Claim 37 further includes an interface configured to output the plurality of areas for presenting the plurality of areas in a first representation, making at least one area of the plurality of areas an active area, and presenting at least one of the at least one active areas in a second representation in response to a user operation on the at least one active area. Paragraph [0089] discloses that as a result of the algorithm of FIG. 7, a plurality of areas is output and these areas can be made active for display on the display in a small representation. Further, upon selection, the areas can be displayed in a large representation.

Independent Claim 37 further defines at least partially dividing at least one page into a plurality of areas to include element-wise rendering of elements contained in the at least one page to obtain a rendered object with a maximum height and a maximum width, checking if a size of the rendered object exceeds a threshold, and forming an area from the rendered object if the threshold is exceeded. In operation 701 of the flowchart of FIG 7, HTML elements of one or several HTML pages may be rendered and investigated in the order they appear in the HTML source code of the page or pages. A calculation of pixel values corresponding to the HTML objects may be, for instance, performed as if an HTML page was shown in its original layout with 100% zoom. As a result, a maximum height and a maximum width in pixels of a number of rendered HTML objects may be obtained. See paragraphs [0073] and [0074]. If the product of

the maximum height and width is larger than a pre-defined threshold, a rectangular area containing the HTML objects rendered in operation 701 is formed in operation 703 of the flowchart of FIG. 7. See paragraphs [0075] – [0077].

The apparatus of Claim 37 is further configured to check if at least one edge of the formed area is not straight and forming a smaller area from the rendered object if at least one edge is not straight. In operation 705 of FIG. 7, the method checks whether the formed area would or would not have a straight top edge. If the formed area would not have a top edge, the algorithm may return to operation 703 and attempt to form a new area with a straight top edge. See at least paragraph [0081].

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

6. ***Grounds of Withdrawal/Rejection to be reviewed on Appeal.***

The following ground of rejection is appealed:

Claims 1, 14, 15, and 25-41 have been rejected under 35 U.S.C. 102(a) as being anticipated by Chen, Y., Ma, W.J., and Zhang, H.J. "Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices," *Proceedings of the 12<sup>th</sup> international conference on World Wide Web (WWW 2003)*, May 20-24, 2003, Budapest, Hungary, previously presented as "Chen."

7. **Argument.**

**Reasons for Requesting Pre-Appeal Brief Request for Review**

These remarks are responsive to the final Office Action dated January 7, 2010. The Final Office Action rejects Claims 1, 14, 15, and 25-41 under 35 U.S.C. § 102(a) as being anticipated by Chen, Y., Ma, W.J., and Zhang, H.J. "Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices," *Proceedings of the 12<sup>th</sup> international conference on World Wide Web (WWW 2003)*, May 20-24, 2003, Budapest, Hungary (hereinafter "Chen"). In view of the following remarks, Appellant respectfully submits that the claims are in condition for allowance.

The Rejection of Independent Claims 1, 15, 32, and 37.

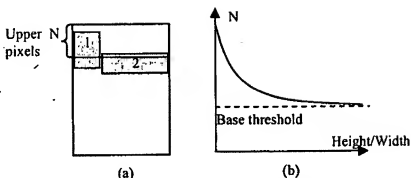
Claim 1 recites, *inter alia* "wherein said at least partially dividing at least one page into a plurality of areas comprises element-wise rendering elements contained in said at least one page to obtain a rendered object with a maximum height and a maximum width, checking if a size of said rendered object exceeds a threshold, and forming an area from said rendered object if said threshold is exceeded[.]" The Office Action cites Chen, page 3, section 3.2.1 as anticipating this claimed feature. The cited disclosure recites "[w]e try to classify a node into one of the header, footer, left side bar and right side bar blocks. If it belongs to none of the above, then we check if it is small enough to be put into the body block. A pair of thresholds (one for width and the other for height) is used to determine whether a node is small enough. If the node exceeds the thresholds, it will be split further. The above process is iterated until all the nodes are classified into the five high-level blocks." Appellant believes that the Examiner has misinterpreted the claimed invention and misapplied the cited art. The claimed invention will "obtain a rendered object with a maximum height and a maximum width, checking if a size of said rendered object exceeds a threshold, and forming an area from said rendered object if said threshold is exceeded[.]" In this regard, forming an area is forming an area for display as outlined in the published application in paragraphs [0073] to [0077] outlining the operations of the flowchart of FIG. 7. Conversely, with regard to Chen, a pair of thresholds is used to determine if a node is small enough and if the node exceeds the thresholds, the node will be split. Then the split nodes will be subject to the iterative process outlined in section 3.2.1 of Chen wherein the node is

classified into one of the five high level content blocks. The method of Chen is clearly distinct from the claimed method, particularly when the claims are given their broadest reasonable interpretation consistent with the specification as outlined in MPEP 2111.

Additionally, Chen analyzes a page and divides it into content blocks, which the Examiner may understand to correspond to the aforementioned feature. However, as described in section 3, Chen divides a page by starting with the whole page and successively dividing the page into individual elements at the next lower level of the hierarchy until the lowermost level is reached. At each level, explicit and implicit separators (described in section 3.3 and 3.4 respectively) are detected and used to split content blocks as described in section 3.1 of Chen. Thus, Chen starts at the topmost level and progressively iterates downwards without knowing beforehand how many hierarchy levels there are to traverse. Consequently, the page division according to Chen is a top-down method and adheres to the layout hierarchy with which the page was created. This is also evidenced by the fact that, according to their respective position in the hierarchy, Chen classifies nodes as header, footer, left side bar, right side bar, or body as described in section 3.2.1.

The above claimed feature, on the other hand, describes a bottom-up approach that groups together the basic elements of a page up to a maximum height and width of the resulting rendered object. This is distinct from the approach of Chen, because according to this feature, the rendering of a group of elements into an object is – at least in this operation – independent of their structure within the hierarchy. Thus, Chen does not anticipate the claimed feature nor is it obvious in view of the disclosure of Chen.

Claim 1 further recites “checking if at least one edge of said formed area is not straight, and forming a smaller area from said rendered object if at least one edge is not straight.” The Office Action cites pages 3-4, section 3.2.2 and FIG. 6(a) of Chen as anticipating this feature. FIG. 6 of Chen is reproduced below.



**Figure 6. Dynamic threshold for header and footer detection**

The Office Action recites “areas 1 and 2 do not form a straight edge, both cannot belong to header, a smaller area 2 is formed to belong to header region.” Appellant asserts that the Examiner has again misinterpreted the teachings of Chen and has improperly applied the teachings of the reference. Section 3.2.2 of Chen is directed to determining which node **corresponds** to the header. The method uses a threshold line at N pixels from the top of the display in which the header must reside. The value of N is variable. The nodes 1 and 2 of FIG. 6(a) do not share a common lower limit to convey the example described in section 3.2.2 in which the straight line, arranged N pixels below the top of the display is varied to determine which node to include in the header. If the node extends below the straight line N, the node is believed to not be part of the header. Thus, **it is irrelevant whether the nodes have straight edges or not** and Chen is silent with respect to the edges of the nodes and their relative shape. Chen further discloses a supplement to the method including factoring in the aspect ratio (width to height) of the node to determine which node is the header. Nowhere in the disclosure of Chen is disclosed any mention of “checking if at least one edge of a formed area is not straight[.]” as recited in the claim. Additionally, the formed areas of Chen (1 and 2 of FIG. 6(a)) are shown as straight which refutes the argument presented by the Examiner. The Office Action also notes that “areas 1 and 2 do not form a straight edge, both cannot belong to the header.” This limitation is not taught or suggested and is contrary to the teachings of Chen. Chen discloses that there may be one or more header blocks in section 3.2 and Chen is silent as to how their edges may or may not align. If the line “N” of FIG. 6(a) were below both of areas 1 and 2, both would be

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

considered to be part of the header and the fact that their lowest lines did not align would be irrelevant.

Because Chen proceeds from the page division given by the predefined layout (see page 3, paragraph 3), Chen does not consider if the resulting division results in straight edges or not. Since the original layout of the page is the authority for splitting the page, Chen is content to adopt the edges of the formed areas from the original layout and accordingly has no reason to deviate from this. Thus, Chen does not teach the aforementioned feature.

For at least the reasons above, Chen does not teach all of the features of Claim 1 nor are these features obvious in light of the teachings of Chen. Therefore, Claim 1 is patentably distinct from Chen and is in condition for allowance. Each of Claims 15, 32, and 37 recite similar features as those discussed with respect to Claim 1 and were rejected together in the Office Action. Thus, in view of the arguments with respect to Claim 1, each of Claims 15, 32, and 37 is patentably distinct from Chen and is similarly in condition for allowance.

The Rejection of the Dependent Claims is Overcome

Because each of the dependent claims includes each of the recitations of a respective independent base claim, Appellant further submits that the dependent claims are patentably distinguishable from the cited references, taken alone or in combination, for at least those reasons discussed above. Accordingly, applicants respectfully submit that the rejections of the dependent claims are overcome and the dependent claims are in condition for allowance.



8. ***Claims Appendix.***

The claims currently on appeal are as follows:

1. (Previously presented) A method comprising:

at least partially dividing at least one page into a plurality of areas,  
presenting said plurality of areas in a first representation,  
making at least one area of said plurality of areas an active area, and  
in response to a user operation on said at least one active area, presenting at least one of  
said at least one active areas in a second representation,

wherein said at least partially dividing at least one page into a plurality of areas  
comprises element-wise rendering elements contained in said at least one page to obtain a  
rendered object with a maximum height and a maximum width, checking if a size of said  
rendered object exceeds a threshold, and forming an area from said rendered object if said  
threshold is exceeded and further comprises checking if at least one edge of said formed area is  
not straight, and forming a smaller area from said rendered object if at least one edge is not  
straight.

2.-13. (Previously cancelled)

14. (Previously presented) A computer-readable medium having a computer program stored  
thereon, the computer program comprising:

instructions operable to cause a processor to perform the method of claim 1.

15. (Previously presented) An apparatus comprising

a processor configured to at least partially divide at least one page into a plurality of  
areas; and

a display configured to present said plurality of areas in a first representation,

the processor being further configured to make at least one area of said plurality of areas an active area; and

the display being further configured to present at least one of said at least one active areas in a second representation in response to a user operation on said at least one of said active areas

wherein the processor being configured to at least partially divide at least one page into a plurality of areas comprises the processor being configured to element-wise render elements contained in said at least one page to obtain a rendered object with a maximum height and a maximum width, being configured to check if a size of said rendered object exceeds a threshold, and being configured to form an area from said rendered object if said threshold is exceeded and further comprises the processor being configured to check if at least one edge of said formed area is not straight and being configured to form a smaller area from said rendered object if at least one edge is not straight.

16.-24. (Cancelled)

25. (Previously presented) The method according to claim 1, wherein said at least partially dividing at least one page into a plurality of areas comprises checking if a formed area can be combined with a previously formed area, and combining said formed area and said previously formed area if they can be combined.

26. (Previously presented) The method according to claim 25, wherein areas are combined if they have a similar width, are horizontally similarly positioned and if their combined size does not exceed a threshold.

27. (Previously presented) The method according to claim 1, wherein said at least partially dividing at least one page into a plurality of areas comprises checking if a formed area has to be re-sectioned, and re-sectioning said formed area if said formed area has to be re-sectioned, wherein said re-sectioning comprises forming a new area which is smaller than the presently formed area.

28. (Previously presented) The apparatus according to claim 15, which apparatus is a mobile phone.

29. (Previously presented) The apparatus according to claim 15, wherein the processor being configured to at least partially divide at least one page into a plurality of areas comprises the processor being configured to check if a formed area can be combined with a previously formed area, and being configured to combine said formed area and said previously formed area if they can be combined.

30. (Previously presented) The apparatus according to claim 29, wherein areas are combined if they have a similar width, are horizontally similarly positioned and if their combined size does not exceed a threshold.

31. (Previously presented) The apparatus according to claim 15, wherein the processor being configured to at least partially divide at least one page into a plurality of areas comprises the processor being configured to check if a formed area has to be re-sectioned, and being configured to re-section said formed area if said formed area has to be re-sectioned, wherein the processor being configured to re-section comprises the processor being configured to form a new area which is smaller than the presently formed area.

32. (Previously presented) A method comprising:

at least partially dividing at least one page into a plurality of areas,  
outputting said plurality of areas for presenting said plurality of areas in a first representation, for making at least one area of said plurality of areas an active area, and for presenting at least one of said at least one active areas in a second representation in response to a user operation on said at least one active area,

wherein said at least partially dividing at least one page into a plurality of areas comprises element-wise rendering elements contained in said at least one page to obtain a

rendered object with a maximum height and a maximum width, checking if a size of said rendered object exceeds a threshold, forming an area from said rendered object if said threshold is exceeded and further comprises checking if at least one edge of said formed area is not straight, and forming a smaller area from said rendered object if at least one edge is not straight.

33. (Previously presented) The method according to claim 32, wherein said at least partially dividing at least one page into a plurality of areas comprises checking if a formed area can be combined with a previously formed area, and combining said formed area and said previously formed area if they can be combined.

34. (Previously presented) The method according to claim 33, wherein areas are combined if they have a similar width, are horizontally similarly positioned and if their combined size does not exceed a threshold.

35. (Previously presented) The method according to claim 32, wherein said at least partially dividing at least one page into a plurality of areas comprises checking if a formed area has to be re-sectioned, and re-sectioning said formed area if said formed area has to be re-sectioned, wherein said re-sectioning comprises forming a new area which is smaller than the presently formed area.

36. (Previously presented) A computer-readable medium having a computer program stored thereon, the computer program comprising:

instructions operable to cause a processor to perform the method of claim 32.

37. (Previously presented) An apparatus comprising

a processor configured to at least partially divide at least one page into a plurality of areas; and

an interface configured to output said plurality of areas for presenting said plurality of areas in a first representation, for making at least one area of said plurality of areas an active

area, and for presenting at least one of said at least one active areas in a second representation in response to a user operation on said at least one active area,

wherein the processor being configured to at least partially divide at least one page into a plurality of areas comprises the processor being configured to element-wise render elements contained in said at least one page to obtain a rendered object with a maximum height and a maximum width, being configured to check if a size of said rendered object exceeds a threshold, and being configured to form an area from said rendered object if said threshold is exceeded and further comprises the processor being configured to check if at least one edge of said formed area is not straight and being configured to form a smaller area from said rendered object if at least one edge is not straight.

38. (Previously presented) The apparatus according to claim 37, which apparatus is a content optimization server.

39. (Previously presented) The apparatus according to claim 37, wherein the processor being configured to at least partially divide at least one page into a plurality of areas comprises the processor being configured to check if a formed area can be combined with a previously formed area, and being configured to combine said formed area and said previously formed area if they can be combined.

40. (Previously presented) The apparatus according to claim 39, wherein areas are combined if they have a similar width, are horizontally similarly positioned and if their combined size does not exceed a threshold.

41. (Previously presented) The apparatus according to claim 37, wherein the processor being configured to at least partially divide at least one page into a plurality of areas comprises the processor being configured to check if a formed area has to be re-sectioned, and being configured to re-section said formed area if said formed area has to be re-sectioned, wherein the

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

processor being configured to re-section comprises the processor being configured to form a new area which is smaller than the presently formed area.

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

9. ***Evidence Appendix.***

None.

In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

10. ***Related Proceedings Appendix.***

None.



In re: Mikko Makela  
Appl. No.: 10/820,442  
Filing Date: April 7, 2004

**CONCLUSION**

In light of the remarks presented herein, Appellant submits that Claims 1, 14, 15, and 25-41 are patentable and the rejections should be reversed.

Respectfully submitted,



Jonathan A. Thomas  
Registration No. 62,200

**CUSTOMER No. 00826**  
**ALSTON & BIRD LLP**  
Bank of America Plaza  
101 South Tryon Street, Suite 4000  
Charlotte, NC 28280-4000  
Tel Charlotte Office (704) 444-1000  
Fax Charlotte Office (704) 444-1111  
LEGAL02/32169570v1

**ELECTRONICALLY FILED USING THE EFS-WEB ELECTRONIC FILING SYSTEM OF THE UNITED STATES PATENT & TRADEMARK OFFICE ON SEPTEMBER 20, 2010.**